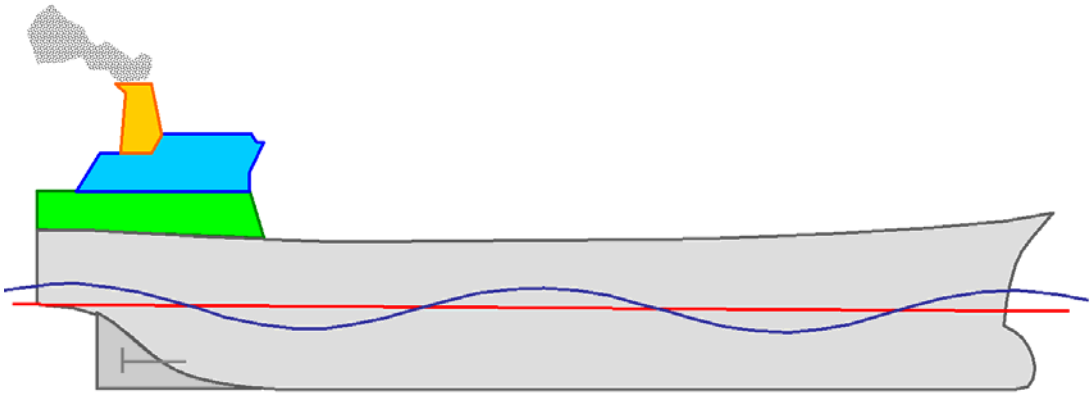
SESAM
USER MANUAL

Wasim



Wave Loads on Vessels
with Forward Speed

DET NORSKE VERITAS

SESAM
USER MANUAL

DET NORSKE VERITAS

SESAM
User Manual

Wasim

Wave Loads on Vessels with Forward Speed

March 28th, 2011

Valid from program version 5.1-01

Developed and marketed by
DET NORSKE VERITAS

DNV Report No.: 2003-0209 / Revision 6, March 28th, 2011

Copyright © 2003 Det Norske Veritas

All rights reserved. No part of this book may be reproduced, in any form or by any means, without permission in writing from the publisher.

Published by:

Det Norske Veritas
Veritasveien 1
N-1322 Høvik
Norway

Telephone:	+47 67 57 99 00
Facsimile:	+47 67 57 72 72
E-mail, sales:	software.sesam@dnv.com
E-mail, support:	software.support@dnv.com
Website:	www.dnv.com

If any person suffers loss or damage which is proved to have been caused by any negligent act or omission of Det Norske Veritas, then Det Norske Veritas shall pay compensation to such person for his proved direct loss or damage. However, the compensation shall not exceed an amount equal to ten times the fee charged for the service in question, provided that the maximum compensation shall never exceed USD 2 millions. In this provision “Det Norske Veritas” shall mean the Foundation Det Norske Veritas as well as all its subsidiaries, directors, officers, employees, agents and any other acting on behalf of Det Norske Veritas.

Table of Contents

1 INTRODUCTION 1-1

1.1 Wasim - 3-dimensional wave loading on ships 1-1

1.2 Wasim in the SESAM System 1-2

1.3 How to read the Manual 1-4

1.4 Status List 1-4

1.5 Terminology and Notation 1-5

1.6 Wasim Extensions 1-5

2 FEATURES OF WASIM 2-1

2.1 Coordinate systems 2-1

2.2 Definition of waves 2-3

2.3 Discretization in space and time 2-4

2.3.1 Time domain simulation 2-4

2.3.2 Spatial discretization 2-7

2.4 Controlling surge, sway and yaw 2-9

2.4.1 Rudder and autopilot 2-9

2.4.2 Soft spring system 2-10

2.4.3 Skeg/passive rudder 2-11

2.5 Roll damping 2-12

2.6 Additional damping and restoring. 2-12

2.7 Spatial filtering 2-12

3 EXECUTION OF WASIM 3-1

3.1 Program Overview 3-1

3.2 Input files 3-4

3.2.1 The Geometry File 3-4

3.2.2 Mass file - alternative form 3-9

3.3 How to run the different modules 3-10
3.4 Output files 3-11
3.4.1 Listing files 3-11
3.4.2 Time domain output 3-12
3.4.3 Frequency domain output 3-13
3.4.4 Load transfer output 3-14
3.5 Program Requirements 3-15
3.5.1 Execution Time 3-15
3.5.2 Memory 3-15
3.6 Program Limitations 3-15

1 INTRODUCTION

1.1 Wasim - 3-dimensional wave loading on ships

Wasim is a program for computing global reponses of and local loading on vessels moving at any forward speed as long as the vessel is not planing. The simulations are carried out in time domain, but results are also transformed to frequency domain by using Fourier transform.

The analysis capabilities of Wasim comprise of:

- Computation of global responses including:
  - — rigid body motions
  - — sectional forces and moments
  - — relative motion at specified points
- Computation of pressure on vessel:
  - — pressure at selected points on the hull
  - — total pressure distribution on the whole hull
- Automatic transfer of normal pressures to a finite element model for structural analysis
  - — frequency domain loads from a linear analysis
  - — time domain snapshots from a non-linear analysis
- Automatic transfer of rigid body accelerations and gravity for computation of inertia and gravity loads in structural analysis
  - — frequency domain RAOs from linear analysis
  - — time domain snapshots from a non-linear analysis

Wasim solves the fully 3-dimensional radiation/diffraction problem by a Rankine panel method. For these methods panels are required both on the hull and on the free surface. Wasim has its own mesh generator hence only the geometry of the hull must be supplied by the user. A geometry model in the required format can be exported from the Nauticus Hull system.

The following non-linear effects are included with the non-linear option:

- Integration of Froude-Krylov and hydrostatic pressure over exact wetted surface.
- Quadratic terms in Bernoulli are included.
- Exact treatment of rotation angles in inertia and gravity terms.
- Quadratic roll damping.

The radiation/diffraction problem is solved on the mean free surface and mean wetted surface with both the linear and non-linear option.

Wasim consists of several different executable elements:

Wasim_Mass	Computes body mass matrix and sectional mass matrices from a given mass model.
Wasim_Setup	Sets up and inverts the influence matrix for the radiation/diffraction problem, solves for the basis flow (i.e. the solution in calm water). The computations in Wasim_Setup must only be redone when the panel model is changed on the hull or on the free surface.
Wasim_Solve	performs the time domain simulation.
Wasim_Fourier	performs the transformation from time domain to frequency domain. This program can analyse time records from many different runs with Wasim_Solve and merge all the data into a single Hydrodynamic Results Interface File (G1.SIF)
Wasim_Stru	load transfer from panel model to FEM model.

1.2 Wasim in the SESAM System

Wasim is an integrated part of the SESAM suite of programs, with interfaces to SESAM Pre and Post processors and the structural analysis module Sestra.

The mass model and the finite element model (i.e. the model to receive hydrodynamic loading) can be given as a SESAM superelement model. Often these are the same model. The geometry input can be exported from the Nauticus Hull system. The Wasim Manager will create the panel model from this geometry input.

The time domain output can be processed by the post processor Postresp_time. The frequency domain output can be processed by the post processor Postresp.

Animation output can be viewed by Xtract.

Figure 1.1 shows the SESAM Overview. A description of the input and output files is given in Chapter 3. See also Figure 3.1.

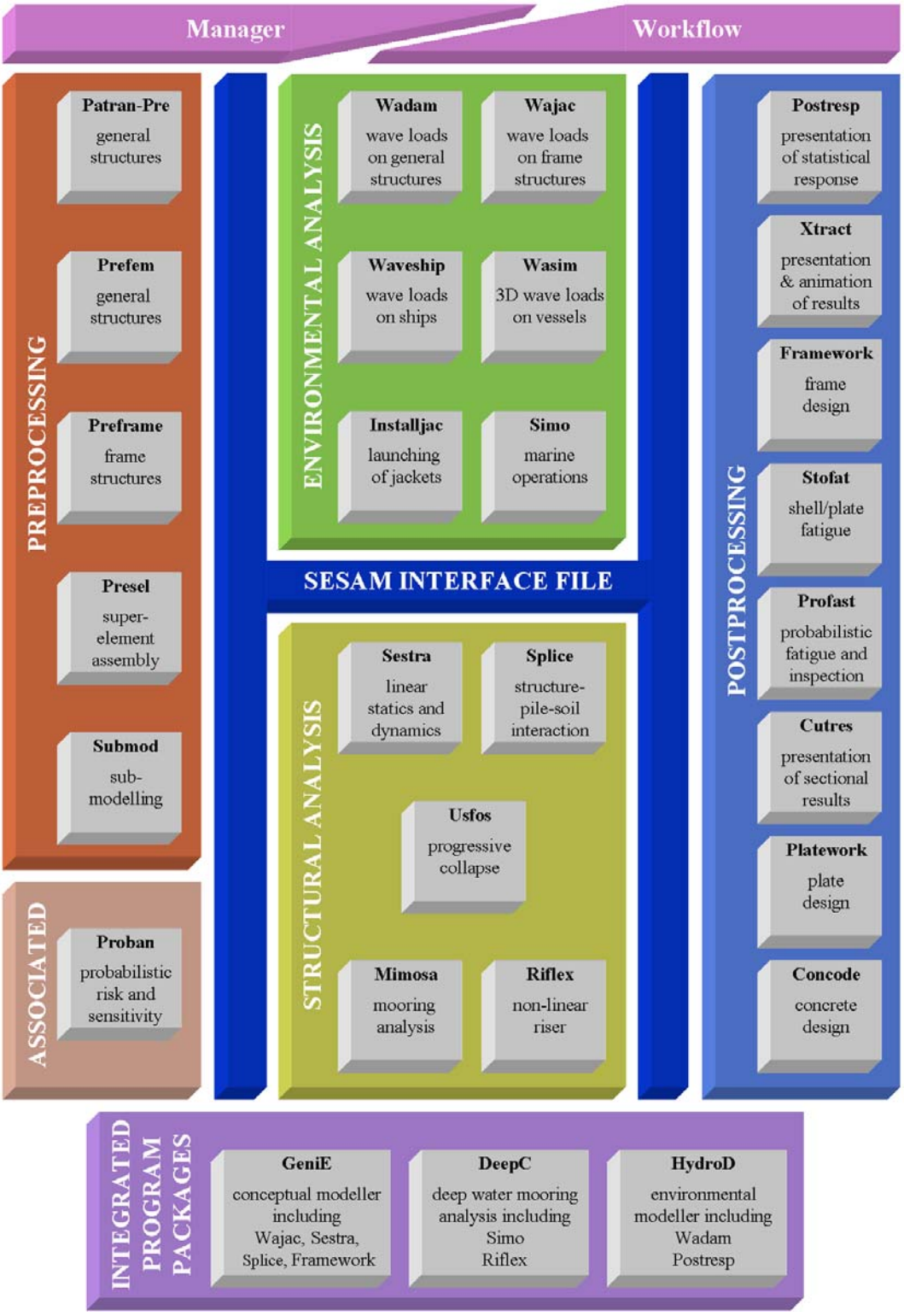


Figure 1.1 SESAM overview

1.3 How to read the Manual

Chapter 2 describes the different features of Wasim.

Chapter 3 contains a description of the data flow in Wasim. This chapter also contains a more detailed description of the input and output files.

A good approach to learn how to use Wasim is to start by studying sections 1.1, 2.1 and 3.1 and then proceed by following the examples described in *HydroD*. The input files to the examples are provided as a part of the installation and are found in the Example directory under the selected installation directory,

After going through these examples, Chapter 2 should be read before you start to analyse your own cases.

1.4 Status List

There exists for Wasim (as for all other SESAM programs) a Status List providing additional information. This may be:

- Reasons for update (new version)
- New features
- Errors found and corrected
- Etc.

The status lists can be accessed on the internet which gives access to the most recent information. The status lists are found on the site www.dnvsoftware.com. Go to the Support tab and click on "SESAM Status Lists". A user name and password are required to get access. These can be obtained from SESAM Support. Send email to Software.Support@dnv.com.

It is also possible to use the program Status for looking up information in the Status List. See the command HELP for how to run Status. However, this program only reads the status lists stored on the local installation.

1.5 Terminology and Notation

Geometry File	input file to Wasim preprocessor containing a sectional description of the vessel.
Section model	model of the mean wetted part of the vessel and the free surface.
Rundata File	input file to *Wasim_Setup* and *Wasim_Solve* containing input/ output file names, control parameters, wave field etc.
Time domain output	the output files from the module *Wasim_Solve*.
Hydrodynamic Results Interface File	output file from *Wasim_Fourier* containing global results and selected pressures in frequency domain. This file can be used as input to the statistical post processor Postresp.
Load file	output file from *Wasim_Fourier* containing rigid body response and pressure distribution on the panel model in frequency domain. Input to *Wasim_Stru*.
Snapshot file	output file from *Wasim_Snapshots* containing rigid body response and pressure distribution on the panel model at selected time steps. Input to *Wasim_Stru*.
Loads Interface File	output from *Wasim_Stru* containing the normal pressure loads on the FEM model. Input to the structural analysis program Sestra and to the animation program Viewer.
Global response	a collective term for rigid body motion, added mass, damping and excitation forces and sectional loads.

1.6 Wasim Extensions

STRU	automatic load transfer to structural analysis programs in frequency domain for linear analysis. This extension consists of the module *Wasim_Stru*.
NL	Non-linear simulation.
NLSTRU	automatic load transfer of non-linear pressure distribution. The pressures are transferred as snapshot of the pressure distribution at selected points in time. The loads will typically be used for a static analysis with Sestra.

2 FEATURES OF WASIM

This chapter describes the features of Wasim. The program system consists of several modules, but a lot of effort has been put into streamlining the analysis process.

The chapter is organised as follows:

Section 2.1 describes the coordinate systems used by Wasim. Section 2.2 provides the definitions of incoming waves and phase angles. Section 2.3 gives a summary of some important issues related to the time domain simulations and the spatial discretization. Section 2.4 and 2.6 describe some of the available motion control systems that can be handled by Wasim.

2.1 Coordinate systems

Wasim uses three different coordinate systems:

The User system	a user defined coordinate system used for the geometry input and other input coordinates. *All input is given in the User system.*
The Global system	used internally for calculations in Wasim. In this system the xy-plane is the mean free surface and x=0 at midship. The transformation from the User system to this system is defined by two translations and one rotation. The translation in x-direction moves the origin to the mid-ship position. Here mid-ship is defined as the mean of two perpendiculars given on the geometry file. The translation in z-direction is given by the draft at midship and a trim parameter defines a rotaton around the y-axis. When transforming from the User system to the Global system we first translate, then rotate.
The Body-fixed system	an internal coordinate system fixed in the vessel. At the starting point of the time simulations the Body-fixed and Global systems coincide.

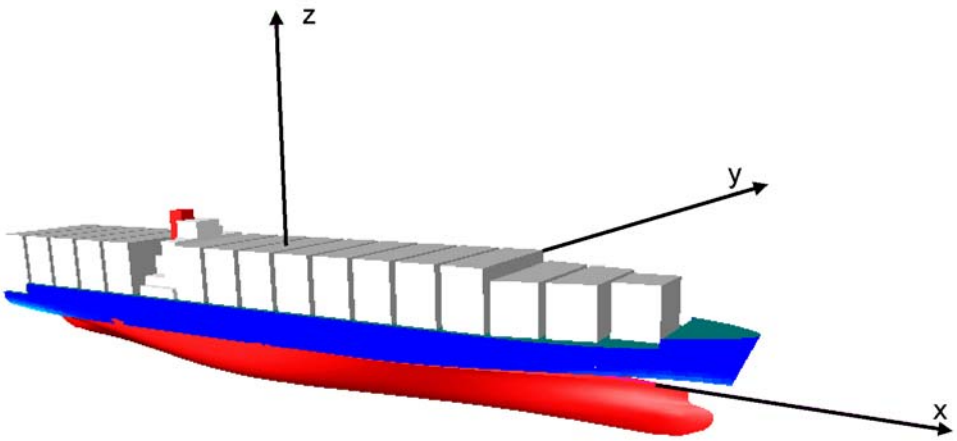


Figure 2.1 Wasim user coordinate system

Limitations on the User system (An example is shown in Figure 2.1):

- If symmetry is to be applied the xz-plane must coincide with the symmetry plane of the vessel.
- The x-axis must point from stern to bow.
- The z-axis must point upward.
- The angle between the xy-plane and the mean free surface should not be too big. There is no strict limit here since the transformation between the User system and the Global system is handled exactly, but we recommend to keep this angle within a few degrees to simplify interpretation of the output.

The User system is the coordinate system used in the geometry input file (see Section 3.2.1). All other input to Wasim is also to be given in the User System.

2.2 Definition of waves

The incoming wave is defined as follows:

$$\eta(x, y, t) = A\cos[(k\cos\beta)x + (k\sin\beta)y - \omega t + \gamma] \tag{2.1}$$

which can alternatively, using complex variables, be written as

$$\eta(x, y, t) = A\exp[i((k\cos\beta)x + (k\sin\beta)y - \omega t + \gamma)] \tag{2.2}$$

k is the wave number. In infinite depth the relation between the wave number and wave frequency (the dispersion relation) is given by:

$$k = \omega^2/g \tag{2.3}$$

Wasim also supports finite depth. In that case the relation is somewhat more complicated.

The wave input to Wasim is given as a sum of harmonic waves of the form given by Equation (2.1). For each wave component the following parameters must be specified:

A	wave amplitude
T	wave period = $2\pi/\omega$
β	wave direction. This is the direction between the positive x-axis and the direction the wave is propagating *towards*. From the definition of the coordinate system we then see that head seas are given by a direction of 180°. 90° will be beam seas with the port side as the lee side.
γ	phase angle

The possibility to define both the wave amplitude and the phase angle for each harmonic component makes it possible to define different realizations of a sea state.

In frequency domain the results are given in the form of transfer functions. The transfer function of a response is the transformation between the input signal and the output signal. The input signal is the incoming wave at the origin of the Global system, i.e. at mid-ship (see Section 2.1) and the output signal is the response history:

Input signal: $A\cos(\gamma - \omega t)$.

Output signal: $AT\cos(\gamma - \omega t - \delta)$.

The transfer function is described by the two parameters T and δ, where T is the ratio of the response amplitude over wave amplitude and δ is the phase lag. A positive phase lag means that the response peak occurs after the wave peak. This is illustrated in Figure 2.2.

When comparing phase angles computed by different applications it must be rememberred that the phase angle as such is a completely arbitrary number, since it depends on the position of the origin in the coordinate system. So the phase lag in itself has no physical meaning. However, relative phase lag between differ-

ent responses do have physical meaning. As an example the phase difference between heave and pitch will be a well defined number for a given vessel, but the phase lag for each of the two motion components will depend on the coordinate system used in the modeling.

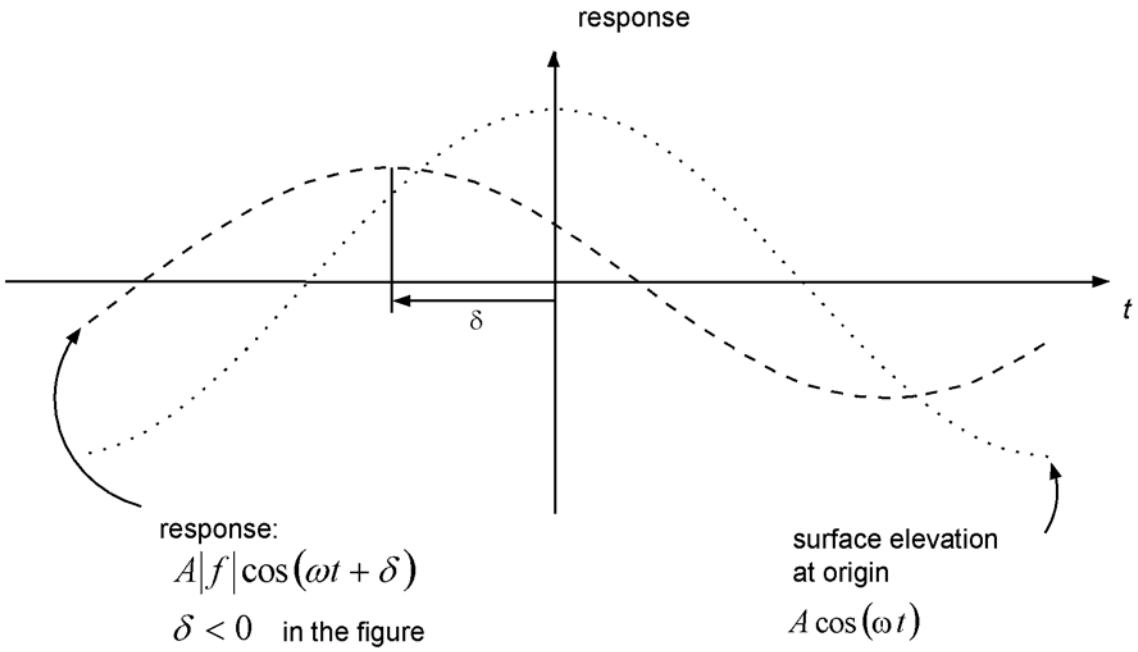


Figure 2.2 Definition of phase lag in the transfer functions

2.3 Discretization in space and time

Wasim solves the wave-vessel interaction problem in time domain. This solution is computed by the module *Wasim_Solve*. This means that discretization in both space and time must be considered in connection with accuracy and stability.

2.3.1 Time domain simulation

An important input parameter to a time domain simulation is the length of the time step. This parameter must be determined from two different aspects: *stability* and *accuracy*. As it will be seen from the discussion below accuracy tends to be the most important limitation in the low to moderate speed cases, whereas stability is the dominating issue in high speed cases. For low to moderate speeds a time step of 0.1s will be small enough in most cases, and it may possibly be increased to 0.15. If in doubt we recommend to do a convergence study before initiating a long run or set of runs.

The time marching schemes used in *Wasim_Solve* are conditionally neutrally stable. This means that the solution becomes unstable if the time step is too large. If the time step is sufficiently small, the solution is neutrally stable, i.e. there is no numerical dissipation. The precise value of the limiting time step depends on

the shape and orientation of the individual panels in the mesh, but the curve shown in Figure 2.3 should be applicable in most cases. The stablity can be analyzed theoretically for a uniform grid with rectangular panels. The presented curve is based on such an analysis with an additional safety margin to account for more irregular grid shapes.

The stability diagram shows the relation between the stability parameter β given by

$$\beta^2 = \frac{h_x}{g(\Delta t)^2} \tag{2.4}$$

and the grid Froude number given by

$$F_h = \frac{U}{\sqrt{g h_x}} \tag{2.5}$$

U is the vessels steady forward speed and h_x is the smallest panel length in the x-direction. The panels used in *Wasim_Solve* are fairly uniform in size so a good approximation to h_x can be obtained by dividing the vessel length with the number of panels in longitudinal direction.

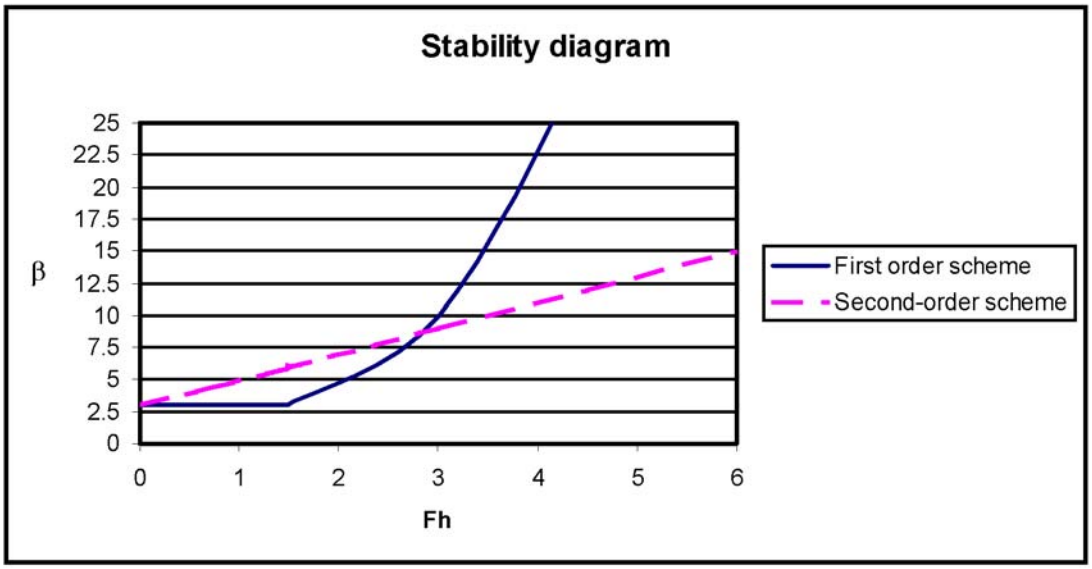


Figure 2.3 Stability diagram for time domain simulation. The stable region is the area above the curves.

There are two different time marching schemes available in *Wasim_Solve*, a first-order scheme and a second-order scheme. The stability limit for both are shown in Figure 2.3. The curves in the figure shows the critical value of β for stable time simulation. The time stepping is stable if the value of β is *larger* than this limiting value.

Example 1: A container ship with length 250m moving with a speed of 20 knots.

A typical discretization will have 40 panels in the longitudinal direction of the ship, giving a value of approximately 6m for h_x. This gives a grid Froude number, F_h=1.3. From Figure 2.3 we then obtain the following stability requirements:

First-order scheme: $\Delta t < 0.3$s

Second-order scheme: $\Delta t < 0.15$s

Although one might assume that the second-order scheme would converge faster than the first-order scheme, practical experience does not show any significant difference in the convergence properties of the two schemes. In the present case, as in most cases for conventional vessels, the limitations on time step for stability are less of an issue than the limitations on the time step for convergence.

Example 2: A high-speed vessel with length 40m moving at 40 knots.

A typical discretization will have 40 panels in the longitudinal direction of the vessel, giving a value of approximately 1m for h_x. For such vessels shorter wave lengths become important and this is why we still need approximately the same number of panels as for the larger ship in example 1. This gives a grid Froude number, F_h=6.4. From Figure 2.3 we then obtain the following stability requirement:

Second-order scheme: $\Delta t < 0.02$s

As can be seen from Figure 2.3 the first-order scheme is not applicable for this problem.

For this case the stability requirement is so strict that convergence is not much of an issue. If the time step is sufficiently small for stability it should also be sufficient for convergence.

In *HydroD* there is an option for the user to select the time marching scheme he wants to use. A typical selection is to use the first-order scheme when the grid Froude number is less than 2.5 and the second-order scheme when it is larger.

The necessary length of the time records will depend on the problem to be solved and (for transfer to frequency domain) the frequencies to be investigated. The length of the time record to be analyzed with the Fourier transform module *Wasim_Fourier* has to be at least

$$T_{min} = \frac{2\pi}{min((\Delta\omega_e)_{min}, (\omega_e)_{min})} + T_{trans} \tag{2.6}$$

where $(\Delta\omega_e)_{min}$ is the smallest difference between two frequencies of encounter in the input dataset, $(\omega_e)_{min}$ is the smallest frequency of encounter and T_{trans} is the length of the transient part of the time series. The length of the transient is typically about 3 times the longest natural period of the rigid body responses. The longest natural period will normally appear for the sway and yaw motions which have to be controlled by an autopilot or a soft spring system (see Section 2.4). A typical transient will be about 5 minutes.

2.3.2 Spatial discretization

The solution method in *Wasim_Solve* is a Rankine Panel method. The difference between a Rankine panel method and a Green's function based panel method (e.g. Wadam) is that the elementary solution in the Rankine method does not satisfy the free surface boundary condition. As a consequence the integral equation to be solved will have unknowns on both the hull and on the free surface. This makes the equation system to be solved larger. On the other hand the computation of the matrices in this equation system is easier than with the full Green's function since the elementary solution is much simpler to compute. Other benefits with the Rankine panel method are:

- Since the free surface condition is not automatically satisfied different free surface conditions can be handled.
- No irregular frequencies

Since the equation system to be solved has unknowns on both the hull and the free surface we must create a mesh on both. An example of a hull grid is shown in Figure 2.4. An example of a free surface grid is shown in Figure 2.5 (complete grid) and Figure 2.6 (close-up of the region close to the vessel). Both the grid on the hull and the grid(s) on the free surface are created in *HydroD* from the geometry input file. There are a number of parameters the user can modify if he wants to adjust the grid.

The extension of the grid depends on the wave frequencies (frequencies of encounter!) in the problem to be solved. In infinite depth the wave length of the radiated and scattered waves is given by

$$\lambda = \frac{2\pi g}{\omega_e^2} \tag{2.7}$$

The extension of the grid must as a minimum be one wavelength of the longest radiated or scattered wave. However, an extension of at least 5 ship lengths will normally suffice even for very small frequencies of encounter. Another range where a large extension of the free surface grid is required is in the region close to the singular point

$$\tau = \frac{U\omega_e}{g} = \frac{1}{4} \tag{2.8}$$

At this point the group velocity of the radiated and scattered waves will be equal to the velocity of the vessel meaning that the energy will not escape. Although this is a physically singular point the solution is well behaved and it can be predicted by Wasim provided that the free surface grid has an extension of about 10 ship lengths.

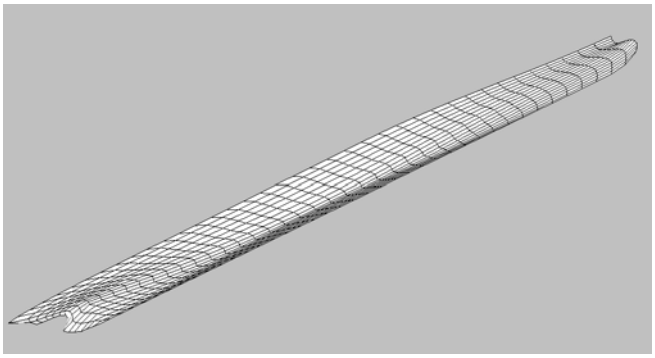


Figure 2.4 A typical hull grid.

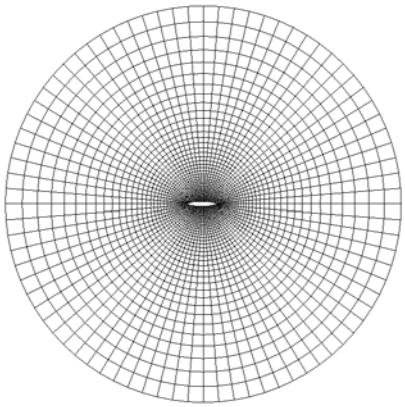


Figure 2.5 A typical free surface grid.

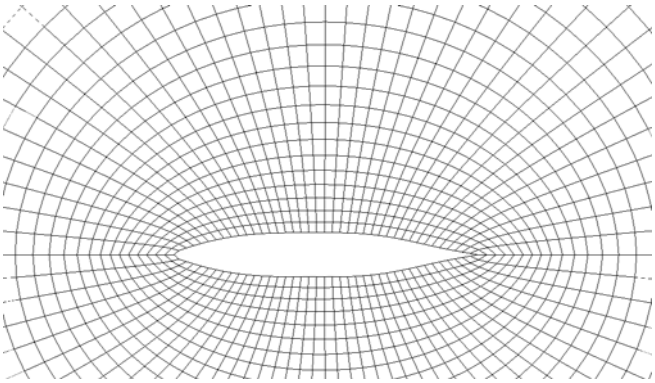


Figure 2.6 A typical free surface grid - close-up.

2.4 Controlling surge, sway and yaw

In a frequency domain program all six degrees of freedom are by definition computed as an harmonic response. In a time domain simulation such as *Wasim_Solve* this is not the case. Since the horizontal reponses have no stiffness there is no mechanism for holding the ship back if it is drifting off. What makes this even more critical is the fact that a ship with no rudder will normally be unstable due to the effect called the Munk moment. Thus the time simulation will also be unstable unless some measure is taken to control the horizontal response. Two different methods for doing this are available in Wasim and will be described in this section.

2.4.1 Rudder and autopilot

The most realistic and intuitive way to control the horizontal motion is to make use of an autopilot and an active rudder. The benefit of this method is that it is close to "real life". The problem with the method is that it can be difficult to find coefficients in the model which makes it stable, and normally no set of coefficients will be stable for all speeds and wave directions. The autopilot should only respond to motions with frequencies around the natural frequencies for sway and yaw, and not to the wave frequency response. This is to some extent handled automatically by the program by using a low-pass filter.

The autopilot model implemented in *Wasim_Solve* has the following form:

$$\delta_R = k_1\xi_6 + k_2\dot{\xi}_6 + k_3\xi_2 + Uk_3\int_0^t \xi_6 dt \tag{2.9}$$

where δ_R is the rudder deflection, ξ_2 is the sway motion, ξ_6 is the yaw motion and $\dot{\xi}_6$ is the yaw angular velocity. For a suitable selection of autopilot coefficients this will give a positionally stable model. With $k_3 = 0$ the model can only be directionally stable.

The total angle of attack on the rudder will be given by the deflection angle and the angle of attack due to the surge, roll and yaw velocities, the yaw angle and the horizontal velocity due to the incoming wave (α):

$$\delta = \delta_R + \xi_6 - \frac{x_R}{U_R}\dot{\xi}_6 + \frac{z_R}{U_R}\dot{\xi}_4 - \frac{1}{U_R}\dot{\xi}_2 + \alpha \tag{2.10}$$

The rudder will contribute to the forces in sway, roll and yaw:

$$F_{2R} = \frac{1}{2}\rho C_L A_R U_R^2 \delta \tag{2.11}$$

$$F_{4R} = -\frac{1}{2}\rho C_L A_R U_R^2 z_R \delta \tag{2.12}$$

$$F_{6R} = \frac{1}{2}\rho C_L A_R U_R^2 x_R \delta \tag{2.13}$$

The different quantities in these formulas are defined as follows:

ρ	Density of water
C_L	Lift coefficient
A_R	Projected rudder area

U_R	Mean velocity on rudder, taken as 1.2**U*, where *U* is the vessel speed
(x_R, y_R, z_R)	Position of rudder
δ	Angle of attack, cf. Equation (2.10)

A total of 6 rudders can be defined in the input. For each rudder the following quantities have to be specified: Position, lift coefficient, projected area, the three autopilot coefficients and the maximum deflection angle.

From the equation system above we see that the rudders will introduce both damping and restoring in sway, roll and yaw. The integral term is included as an excitation force.

Notice that the motions internally in Wasim are computed with respect to the origin in the Global system. Since this origin is at midship the rudder will always give a yaw moment.

Non-linear modification:

In a non-linear analysis the quantity U_R in Equation (2.11) - (2.13) is computed as the total velocity on the rudder. Thus we add the components due to the rigid body motion and the incoming wave to the mean velocity 1.2**U*. This also means that we introduce non-linear damping and restoring in surge, sway, roll and yaw.

2.4.2 Soft spring system

An alternative to using the autopilot to control sway and yaw is to use an artificial spring system, i.e. to introduce an artificial stiffness in the horizontal modes. In addition artificial damping can also be specified for these modes in order to make the transient motions die out. For surge this is the only available control mechanism.

The spring system used is shown in Figure 2.7. The springs are attached at the bow and stern of the vessel ("old" model) or only at the stern ("new" model), and designed as shown in the figure to decouple surge from sway and yaw. Both stiffness and damping are defined for the springs. The user does not specify the damping and stiffness coefficients directly. Damping is given as fraction of critical damping. The stiffness is implicitly given by specifying the natural period in the modes to control. Notice that the user can specify data for surge and for sway *or* yaw. *If data are specified for both sway and yaw the program will only use the data for yaw.* The natural periods specified should be much longer than the natural period in roll in order to avoid unwanted interference with the roll motion. Typical values are in the range 60-120s for conventional vessels, 30-60s for high speed vessels. Notice that increasing this value may also increase the length of the transient in the computed time series, cf. Section 2.3.1. The stiffness will be computed by assuming that the modes are uncoupled. Thus the actual natural periods may differ somewhat from the periods given by the user.

In connection with an autopilot model it is not natural to define any springs for control of sway and yaw (although this is possible). A spring for surge control will still be needed since this mode is not controlled by the autopilot.

Note: If the force from the springs is not included in the computation of the global loads these loads will not be exactly zero when integrated over the whole vessel.

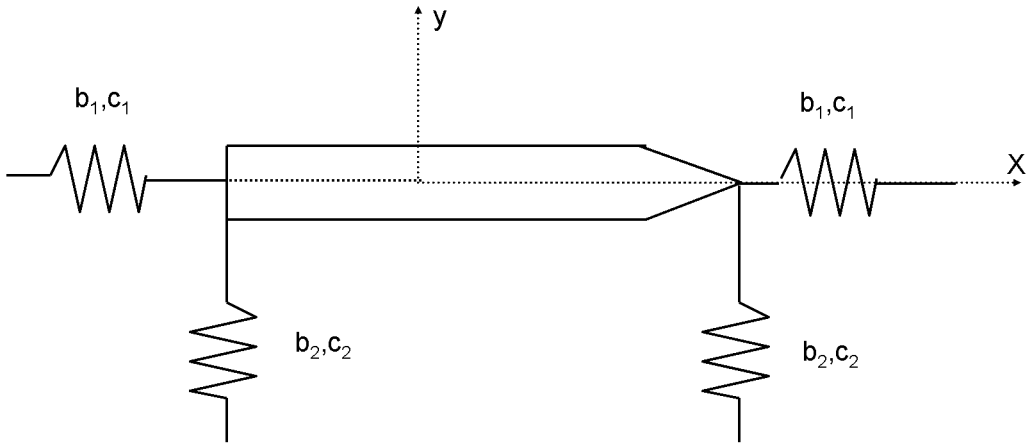


Figure 2.7 The soft spring system.

2.4.3 Skeg/passive rudder

A ship moving with forward speed in calm water is likely to be unstable in yaw. The destabilizing effect is called the Munk moment. This moment is caused by the unsymmetric flow around the ship when the yaw angle is non-zero. A passive rudder or skeg can be used to correct for this destabilizing moment.

The restoring coefficient of the Munk moment can often be approximated by the added mass coefficients in surge and sway by the formula:

$$C_{66}^{Munk} = -U^2(A_{22} - A_{11}) \tag{2.14}$$

where U is the vessel speed. This will always be a negative number.

The restoring coefficient of the skeg is given by:

$$C_{66}^{Skeg} = -\frac{1}{2}\rho C_L A_R U_R^2 x_R \tag{2.15}$$

The different quantities in this equation are described in Section 2.4.1.

The position of the skeg is normally just in front of the real rudder. After defining the position and lift coefficient of the skeg the projected area can be determined from the requirement that

$$C_{66}^{Skeg} + C_{66}^{Munk} \geq 0 \tag{2.16}$$

A skeg will only give directional stability so it may be useful to combine this with a soft spring system (cf. Section 2.4.2). This is even more important since stability in calm water does not necessarily mean that the vessel is stable in waves.

The user input for defining a skeg is exactly as the user input for the active rudders with all three autopilot coefficients given as zero.

2.5 Roll damping

The roll damping model in Wasim is a quadratic damping model:

$$B_{44} = B_1 + B_2\left|\dot{\xi}_4\right| = \left(b_1 + b_2\frac{\left|\dot{\xi}_4\right|}{\omega_4}\right)B_4^{crit} \tag{2.17}$$

B_{44} is the roll damping coefficient. B_1 is the linear damping coefficient and B_2 is the quadratic damping coefficient. B_4^{crit} is the critical damping in roll. The user specifies the non-dimensional coefficients b_1 and b_2. The linear term is given by the user as fraction of critical damping. The quadratic term is given in the unit 1/degrees (i.e. the roll velocity in the above formula is in degrees/s).

In the linear version of Wasim only the linear term is used even if both coefficients are specified in the input.

It should be noticed that the motion control systems described in Section 2.4 will also give roll damping. This roll damping will be added to the roll damping coefficient specified by the user.

2.6 Additional damping and restoring.

In addition to the specification of damping in connection with the spring model (Section 2.4.2) and the roll damping (Section 2.5) additional damping and restoring can be specified for all modes of motion. The user can specify a full 6x6 damping matrix and a full 6x6 restoring matrix. Alternatively the damping for each mode (i.e. diagonal in damping matrix) can be specified relative to critical damping. When this is recomputed to actual damping inside the program the hydrostatic restoring, restoring due to gravity and additional restoring specified by the user is used to compute the critical damping for each mode. Restoring from the spring model and autopilot is not included in the computation of the critical damping. The damping and restoring specified in this way is always added to all other damping and restoring components that are computed in the program. If damping relative to critical damping is specified for a mode with no restoring the damping will also be set to 0 for this mode.

2.7 Additional damping force in ramp_length

Additional damping forces during the ramp_length are introduced in three horizontal directions, i.e. surge, sway and yaw. This function will be switched on only if users have allowed smooth set up (i.e. ismooth=1 and ramp_length >0).

Coefficients of dampramp(1:6) indicate the damping at six directions in the interval time=[0, 0.7*ramp_length] and will decrease gradually to zero as time approach to ramp_length. The default settings are dampramp=1.0,1.0,0,0,0,1.0. This means dampings equal to the critical damping in surge, sway and yaw are used to calculate damping force during ramp_length.

Users can switch off damping force completely by setting dampramp = 6*0.0 in Extra Parameter of WasimActivity.

2.8 Spatial filtering

In the numerical solution wave lengths around 3 panel sizes ($3\Delta x$) will have a group velocity equal to the vessel velocity. This is a consequence of the numerical methods used and do not represent any real life physics. This is a potential source of instability. For this reason spatial filtering has to be applied at regular intervals (specified by the user). The filter will reduce the energy of wavelengths shorter than 5 panel sizes. If there is no input on the frequency corresponding to this critical wave length, occasional filtering is sufficient. If there is input on this frequency filtering every time step may be needed. In extreme cases (typically in the high speed range) this may even impose restrictions on the size of the time step.

3 EXECUTION OF WASIM

This section provides information on:

- How to run Wasim
- Input and output files
- Program requirements
- Program limitations

3.1 Program Overview

The flowchart of a linear analysis with Wasim is shown in Figure 3.1. The figure also illustrates the interface to Nauticus and other SESAM modules. The blue box contains all the Wasim modules. Everything outside this box is not a part of Wasim.

The contents of the blue box is expanded in more detail in Figure 3.2. This shows the structure of the Wasim program system. In an actual analysis case the user works from top to bottom. Before performing the Export operation all input data must have been defined.

Output of the Load Interface files requires the STRU extension for frequency domain loads and the NLSTRU extension for snapshot loads.

The fourier transform step is mainly used in a linear analysis.

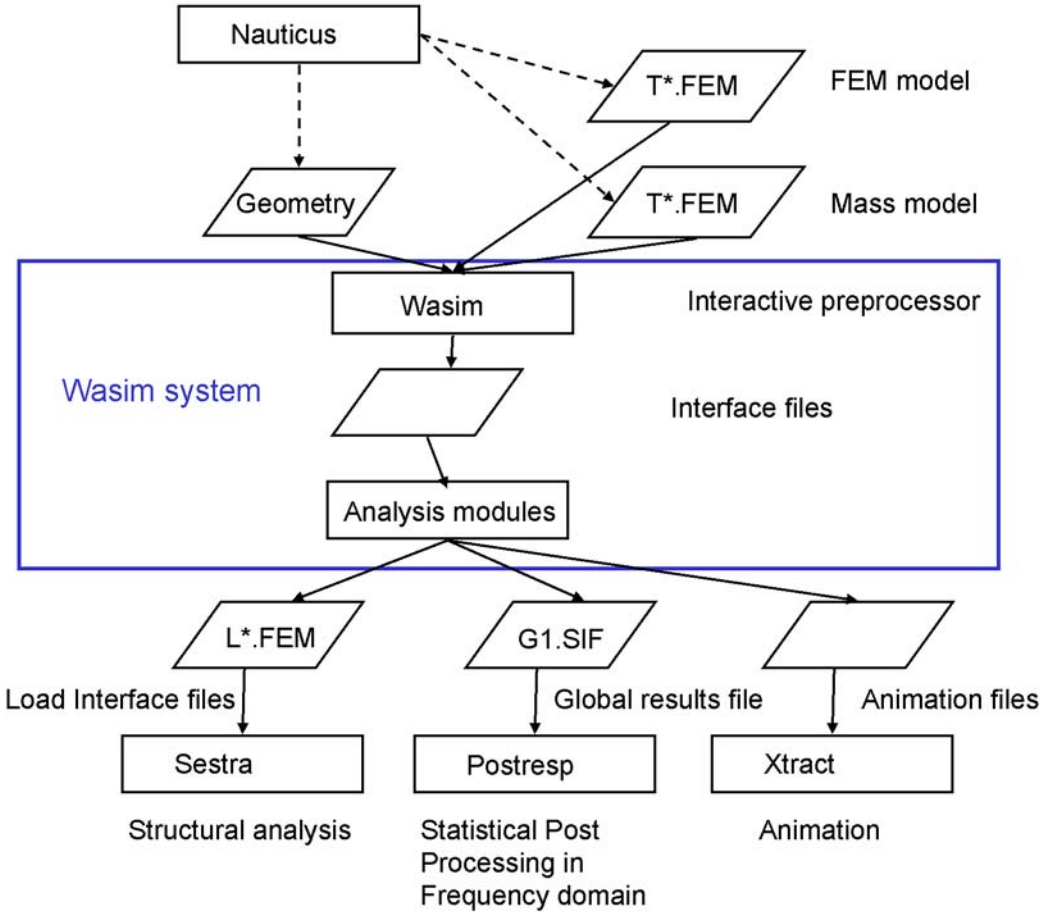


Figure 3.1 The program environment for Wasim in linear mode.

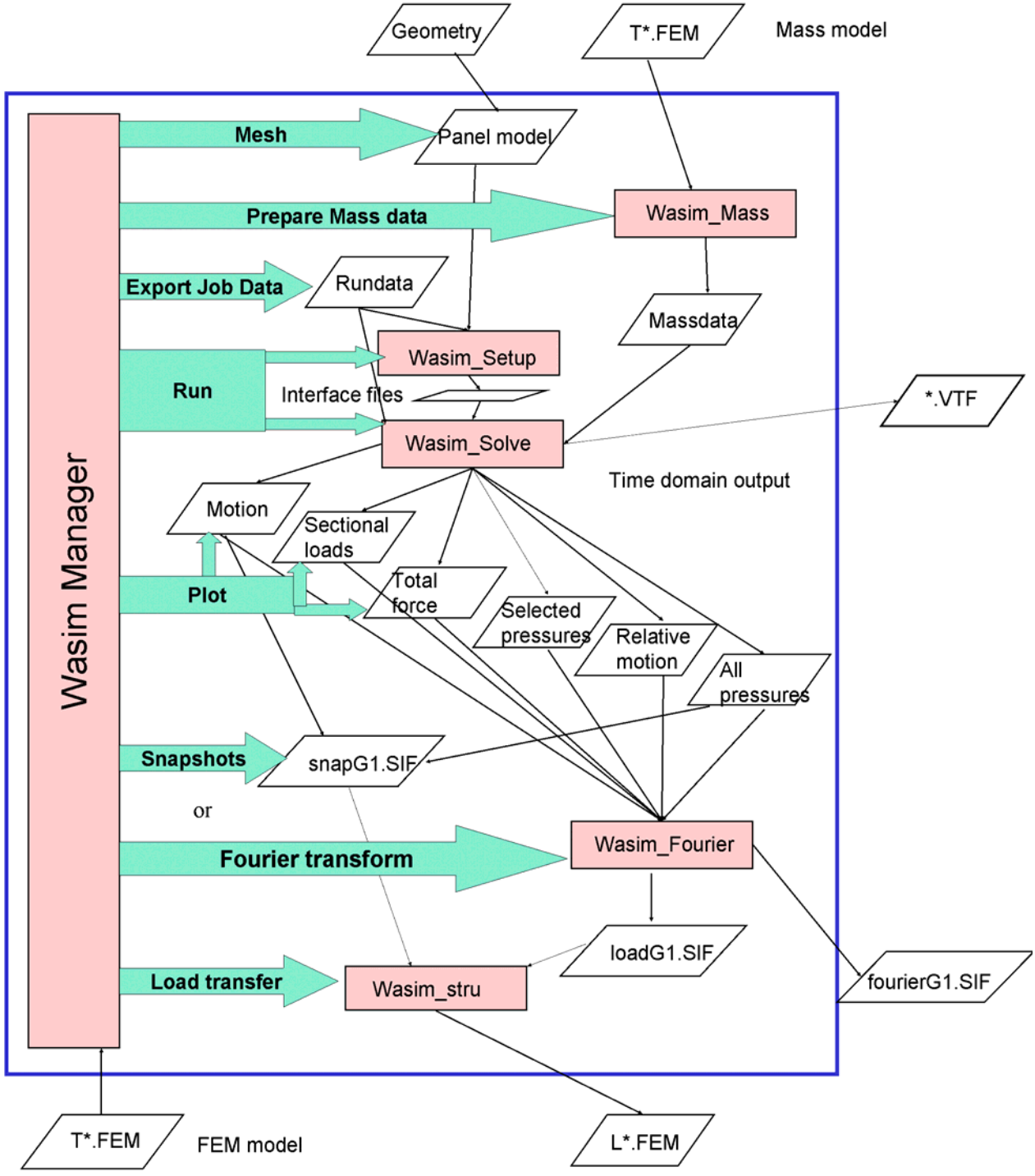


Figure 3.2 The Wasim system in more detail. The blue box is the same as in Figure 3.1. Red boxes are different programs in the Wasim system. Green arrows show commands to be executed in the Wasim Manager.

3.2 Input files

All the necessary input files may be exported from Nauticus. Alternatively the FEM and Mass model can be created by the standard SESAM preprocessors GeniE, Patran-Pre and Presel. The mass model also has an alternative form where the mass is described as a set of point masses. The FEM model is needed only for load transfer.

Instead of a file containing the mass distribution the global mass data (total mass, radii of gyration, center of gravity) can be given as direct input in *HydroD*. This is sufficient if sectional loads are not wanted. Computation of sectional loads will require more detailed mass information.

All the remaining input data are given interactively while running *HydroD*.

The Geometry File, the FEM model and the Mass model must all be in the same frame of reference. All other input coordinates given in *HydroD* must also refer to this frame of reference, the User system. See Section 2.1

3.2.1 The Geometry File

The Geometry File contains a description of the vessel geometry in the form a set of hull parts. Each part is described by a set of offset points. The offset points are sorted into sections and each section is described by a series of points. The sections need not be plane curves. On all hull parts the sections must be given *from bow to stern*. The first section describes the bow shape, the last section describes the stern shape. These two curves are allowed to intersect other curves, but the intermediate sections cannot intersect each other. For the bow and stern curves an additional limitation is that although the lower part of the curve may be downstream (bow curve) or upstream (stern curve) of any point on any other curve, the upper part can not. Thus if you follow the bow curve starting from the keel, when you get to a point which has a larger x-coordinate than any other curve, all the remaining points on the bow curve must also have larger x-coordinates than any other curve. Similarly for smaller x-values on the stern curve.

A vessel can be described by a set of different parts, but *it is recommended to use as few parts as possible* while maintaining a mesh of good quality. *The curves can only intersect the free surface once.* This means that side hulls on a multi-hull vessel must be split in two parts: the part outside the keel line and the part inside the keel line. For both parts the offset points for each section starts at the keel line. The same procedure is needed for a non-symmetric monohull.

There are four different part types that can be specified:

Surface piercing parts defining a waterline.

These are parts that may intersect the free surface. If they do intersect, the intersection point will be a part of a waterline, i.e. it will be on one of the boundary curves on a mesh on the free surface. This type of parts are typically used for the outside of a monohull, or both sides on side hull (where there is to be a free surface mesh in the region between the hulls). On such parts there will be one MxN mesh on the wetted area and another mesh on the "dry" area (the area above the mean waterline).

There are two limitations in the ordering of such hull parts:

All hull parts defining the same waterline must be defined before parts defining another waterline.

The waterlines must be defined from port to centerplane/starboard.

An example: On a trimaran all the parts on the outside of the side hull must be specfied before all parts defining the inside of the side hull. These must again be given before all parts defining the main hull. However, all the parts defining the same hull side can be given in any order (cf. Figure 3.3).

The points on each section must be given *from keel to deck*.

Surface piercing parts that do not define a waterline.

Such parts are typically used on e.g. a twin-screw hull for the area inside the skegs, if this part of the hull may be surface piercing. If such a part of the vessel were included in a waterline the corresponding free surface mesh may be totally deformed and make the analysis fail. Another typical use of this part type is for the transom stern if a mesh is wanted here (typically at small speeds), while still avoiding that the free surface mesh must curve around the sharp corner at the edge of the transom.

Parts of this type need not be given in any particular order.

The points on each section must be given *from keel to deck*.

Totally submerged parts.

These are parts that are totally submerged in all loading conditions. Totally submerged parts do not have to be given in any particular order. The sections must (as for all part types) be given from bow to stern. The points on each section must be given in clockwise order when viewing towards the positive x-axis. This means that when you move from the first point towards the last (looking towards the bow) the wet side is on the left hand side.

Totally "dry" parts.

These are parts that are totally above the mean free surface in all loading conditions. Such parts can also be specified in any order. The sections must be given from bow to stern and the points must be given in such an order that when you move from the first point towards the last (looking towards the bow) the potentially wet side is on the left hand side. Thus if this part is e.g. the tunnel between two hulls on a catamaran the points should be given from center towards port (increasing y-values). If this part is e.g. a deck the points should be given from port towards center (decreasing y-values).

Note: Parts defined as surface-piercing may be totally submerged or totally dry in some loading conditions, whereas parts defined as totally submerged must be totally submerged for all loading conditions and parts defined as totally "dry" must be totally "dry" for all loading conditions.

In principle it is possible to identify all parts as surface piercing. A benefit of using totally submerged or totally dry parts is the option to make the parts from closed curves. This will not be possible with surface-piercing parts since a curve cannot intersect the free surface more than once.

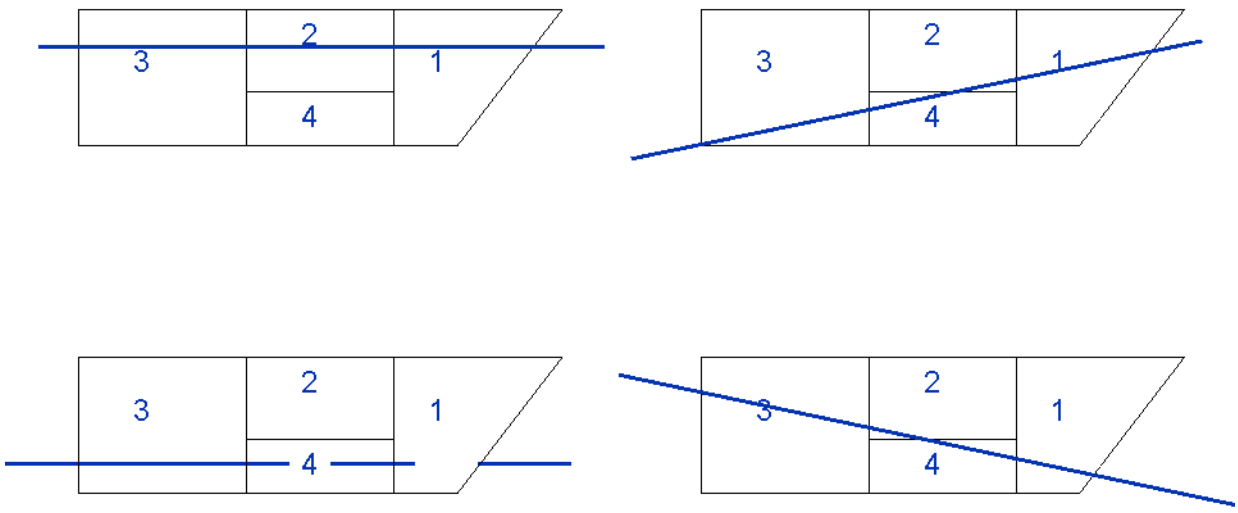


Figure 3.3 The figure shows the possibilities of a geometry model with several parts. If all four parts are specified as surface-piercing all the indicated waterlines can be handled. In the upper left figure the waterline is defined by the parts 1, 2, 3 in this order. In the lower left it will be defined by the parts 1, 4, 3 in this order. Simlarly the ordering will be 1, 2, 4, 3 in the upper left case and 1, 4, 2, 3 in the lower left case. In Wasim version 3.4 all these are accepted. If the starboard side is also modelled with similar parts it is however required that those parts are numberred from 5 to 8. The part number is not given on the geometry file, but is defined implicitely by the sequence of the parts on the file.

The format of the Geometry File is as follows:

<Text>
NPARTS
IWET(1) .... IWET(NPARTS)
ISYM
AP FP
<Dataset no. 1>
.....
<Dataset no. NPARTS

where

<Text>	Descriptive text
NPARTS	No. of hull parts
IWET(i)	Identification of wet side for hull part no. i
	IWET(i) = 0: this hull part is totally submerged

	IWET(i) = -1: starboard side of waterline is wet
	IWET(i) = 1: port side of waterline is wet
	IWET(i) = 2: this hull part is totally "dry", i.e. above the mean waterline.
	IWET(i) = -10: starboard side of part is wet. Part do not define a waterline.
	IWET(i) = 10: port side of part is wet. Part do not define a waterline.
ISYM	Symmetry flag
	ISYM = 0: whole vessel modelled,
	ISYM=1: only port half modelled
AP FP	x-coordinates defining aft and fore perpendiculars
Dataset no. i	Each dataset is a block of data as described below.

Each of the datasets (hull parts) have the following format:

```
<Text>
NSECT
NPOINT(1)  <Section ID>
X(1,1),Y(1,1),Z(1,1)
......
X(NPOINT(1),1),Y(NPOINT(1),1),Z(NPOINT(1),1)
......
......
NPOINT(NSECT) <Station ID>
X(1,NSECT),Y(1,NSECT),Z(1,NSECT)
......
X(NPOINT(NSECT),NSECT),Y(NPOINT(NSECT),NSECT),Z(NPOINT(NSECT),NSECT)
```

where

NSECT	Total number of sections.
NPOINT(i)	Number of offset points in section no. i
<Station Id>	Text describing the section (optional but recommended).
X(j,i),Y(j,i),Z(j,i)	Coordinates of offset points.

Example 1: Monohull

For a symmetric monohull only one hull part is often used. In such a case the first five lines on the Geometry File will be

Tanker
1
1
1
0.0 275.0

followed by one datset with offset points.

Example 2: Trimaran

For a trimaran it is reasonable to have three hull parts. Due to the required order of parts part 1 will be the outside of the side hull. This part is wet on the port side. Part 2 will be the inside of the side hull. This part is wet on the starboard side. Part three will be the port half of the center hull, which is wet on the port side, cf, Figure 3.4. Thus the first five lines on the Geometry File will be

Trimaran example
3
1 -1 1
1
0.0 45.0

followed by three datsets with offset points.

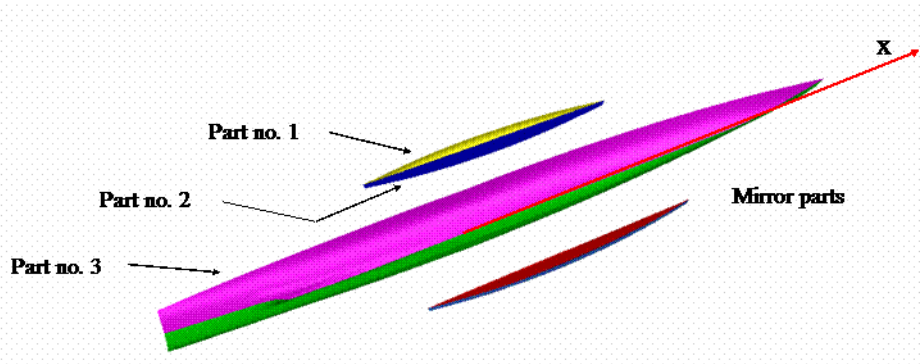


Figure 3.4 Trimaran example.

Example 3: Swath

For a Swath with two struts on each side it is reasonable to have five hull parts. Due to the required order of parts part 1 will be the outside of the "bow" strut. This part is wet on the port side. Part 2 will be the outside of the "stern" strut. This part is also wet on the port side. Part 3 and 4 will be the inside of the "bow" and "stern" struts respectively. These parts are wet on the starboard side. Part 5 will be the pontoon which is totally submerged, cf. Figure 3.5. Thus the first five lines on the Geometry File will be

Swath example
5
1 1 -1 -1 0
1
-25.0 25.0

followed by five datsets with offset points.

For totally submerged parts like the pontoon section for the Swath the point sections must still be given from bow to stern, but ordering from keel to waterline does not have any meaning. For such parts the point must be ordered such that the cross-product of a vector pointing from bow to stern and a vector pointing from one point to the next gives a vector pointing into the fluid. For the pontoon of the swath this means that *the points around the contour must be given in the clockwise direction when looking towards the bow*.

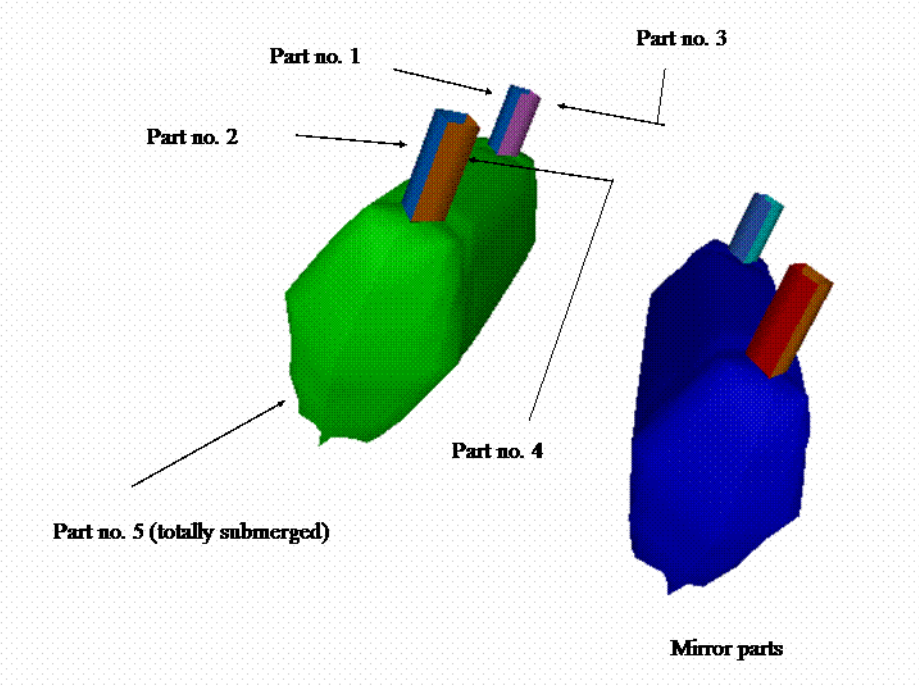


Figure 3.5 Swath example.

3.2.2 Mass file - alternative form

The mass model can be given in the form of a SESAM superelement model (T-file). Often the mass model and FEM model will be identical. Wasim also supports an alternative formulation. The format of this file is as follows:

<Text>
NPOINTS
SCALEM SCALEX SCALEY SCALEZ
M(1) X(1) Y(1) Z(1)
.....
M(NPOINTS) X(NPOINTS) Y(NPOINTS) z(NPOINTS)

where

<Text>	Identification text
SCALEM SCALEX SCALEY SCALEZ	Scaling factor for mass and coordinates, applied to all mass points
M(i)	Mass of point no. i.
X(i) Y(I) Z(I)	Coordinates of point no. i

The scaling parameters are included for easy change from one scale to another with the same mass distribution, e.g. from model scale to full scale.

There is no symmetry option in the mass file. *Symmetry is handled implicitly.* If all y-coordinates are non-negative *HydroD* will assume that symmetry is to be used, and it will automatically add a mirror image of all points with non-zero y-coordinate. If at least one mass point with negative y-coordinate is found it is assumed that the file contains the complete mass distribution.

It is recommended to use the file extension *.mass* or *.mas* for the point mass file. These are the file extensions that automatically appear in the browser.

3.3 How to run the different modules

HydroD is a graphical interactive pre-processor which is started by the desktop icon or from the Start menu. In this module the vessel and the panel model are defined and all analysis parameters are set up. The module exports all files needed for running the programs *Wasim_Setup, Wasim_Mesh/Hydro_Mesh, Wasim_Solve, Wasim_Fourier* ,*Wasim_Snapshots* and *Wasim_Stru*.

Wasim_Fourier can also be used for a non-linear run with a single harmonic wave component. In this case the program will find the contributions from the different harmonic components.

Note: For load transfer in frequency domain all runs included in a Wasim_Fourier analysis must have the same speed since Wasim_Stru only handles one speed. In practice this means that for load transfer Wasim_Fourier is most typically run on the Job level.

When *Wasim_Fourier* performs a fourier analysis of a run it reads the wave periods from the input file to *Wasim_Solve*. Then it computes the corresponding frequencies of encounter. These are the frequencies that are present in the time series to be analyzed. *Wasim_Fourier* then computes the contribution from each of those components and outputs it as the contribution for each wave period. This will only work if there is a one to one correspondence between the frequency of encounter and the wave frequency, since it will not be possible to distinguish the contributions from two different wave periods with the same frequency of encounter. This relation between frequency of encounter and wave frequency is given by the formula

$$\omega_e = \omega - Uk\cos\beta \tag{3.1}$$

where β is the wave direction. For waves from behind (i.e. wave directions between $-90°$ and $90°$) in infinite depth (cf. Equation (2.3)) this relation will look as shown in Figure 3.6. If all the wave periods in a run are inside one of the three intervals between (or to the right of) the vertical red lines we are guaranteed a

unique relation between wave frequency and frequency of encounter. Hence each of these three intervals are handled in separate runs.

For wave directions from head to beam seas this problem do not occur since these headings will always give a one to one relation between the two frequencies.

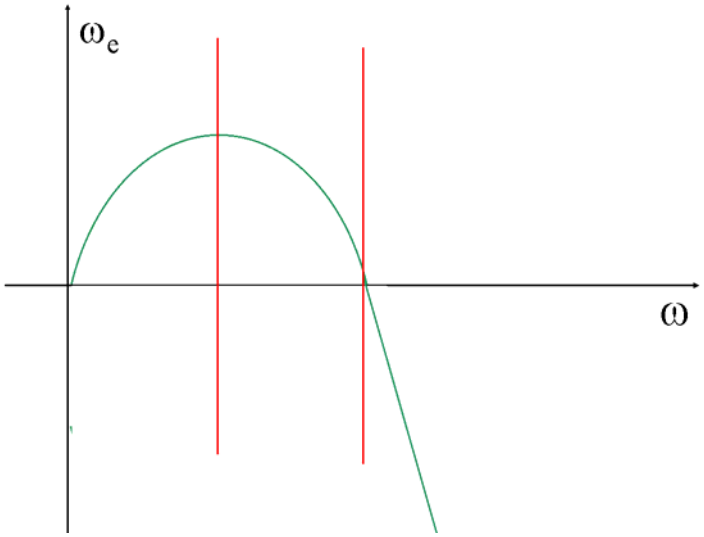


Figure 3.6 The relation between frequency of encounter and wave frequency in following sea.

Wasim_Stru picks up the Load file created by Wasim_Fourier or by a snapshot collection. It then converts this to a set of load cases on the FEM model for further analysis by Sestra. Since Wasim_Stru only handles one speed in the Load file it is most typically executed on the Job level, although it may also be executed on the Loading condition level. This could be relevant if there are different jobs for different headings, but all with the same speed.

3.4 Output files

The output files are written to a directory structure in the current workspace.

3.4.1 Listing files

A typical linear run with Wasim will include one (or more) runs of *Wasim_Setup*, several runs of *Wasim_Solve*, one run of *Wasim_Fourier* and (if load transfer is wanted) one run of *Wasim_Stru*. The data-flow between the modules is shown in Figure 3.2.

A non-linear analysis will normally only include one (or more) runs of *Wasim_Setup*, one (or more) runs of *Wasim_Solve* and one (or more) runs of *Wasim_Stru.* If Wasim_Solve is run with one harmonic wave only (e.g. a design wave) it is also possible to run Wasim_Fourier to obtain the values for the different harmonic components.

All the modules write a listing file with some general information about the runs. The listing file from *Wasim_Fourier* also contain some numerical results. The listing file from *Wasim_Stru* contains valuable information of the results from the load transfer (in particular the no. of elements without load). For all the modules the name of the listing file will be *<name>.LIS* when the input file is *<name>.inp*. For Wasim_Setup and Wasim_Solve <name> will be the name of the run, for Wasim_Fourier it will be *fourier* and for Wasim_Stru it will be *stru*.

In addition to the listing file *Wasim_Solve* writes a file with characteristic data for the vessel. The file contains the global characteristics of the vessel and a number of matrices (e.g. mass and restoring matrix). The name of this file is *<Name of job>.par*.

3.4.2 Time domain output

Wasim_Solve produces several output files containing time records. The files are formatted such that they can be read by the time domain post processor Postresp_time. All files have a data format as follows:

<Some text lines>

$t_1 \quad x_1(t_1) \quad x_2(t_1)$ ..... $x_n(t_1)$

$x_{n+1}(t_1)$ .....

$t_2 \quad x_1(t_2) \quad x_2(t_2)$ ..... $x_n(t_2)$

$x_{n+1}(t_2)$ .....

.....

.....

If there are more columns for each time step than can be fitted on a single line, continuation lines will start with a blank field of at least the same size as the time column.

The following files with time records can be written:

a Rigid body motion in 6 degrees of freedom
This file contains three text lines on top of the file and seven data columns for each time step. All columns are on the same line. The first six are the rigid motion at the motion reference point, the seventh is the time trace of the incoming wave at the origin of the Global system (cf. Section 2.1). *For a non-linear run this origin will follow the translations of the vessel*, so in this case this column is not the time trace of the incoming wave at a fixed point on the free surface.

b Force and moment on vessel
For freely floating vessel this file contains the total force and moment (i.e. the right hand side in Newtons law). For the fixed vessel case the file contains the diffraction force (including Froude-Krylov component). Hence from Wasim_Fourier the output will be the excitation force transfer function. For the forced motion case the file contains the radation force and hence the fourier analysis will give the added mass and damping. The data format is the same as the motion files, except that the first three columns are the total force components and the next three are the total moments with respect to axes through the motion reference point. For the case of a forced motion the seventh column will contain the time history of the forced motion. Otherwise it contains the time trace of the incoming wave as for the motion file.

c Sectional loads on sections normal to x-axis
One text line on top of the file. Column 1-6 is the first section, column 7-12 is the second section and so on. There are two sections (i.e. 12 columns) on each line.

d Sectional loads on sections normal to y-axis
Same format as for x-cuts.

e Sectional loads on sections normal to z-axis
Same format as for y-cuts.

f Relative motion between vessel and free surface
One text line on top of the file. One column per point. Maximum 7 points (columns) per line.

g Pressure at selected positions
Same format as relative motions.

h Rudder motion
First column: Angle of rudder one. Second column: Sway response at the intersection of the z-axis and the free surface. Third column: Yaw response in degrees. Fourth column: Force in y-direction on rudder one. Column 5-6: Angle of and force on rudder two. Column: 7-8: Same data for rudder three and so on for the remaining rudders. All data on one line. (Maximum number of rudders is 6.)

i Pressure on all panels
The first line contains the text "Panels and steps" followed by the number of panels on the whole hull (i.e. including mirror image) and the number of time steps in the file. The data section has one column for each panel. Maximum 100 columns per line.

In addition to these ASCII files Wasim_solve also writes a binary file with data which can be read into Xtract for animation of the vessel motion, surface elevation and pressure on the hull.

Note: For all the vector data in the time domain output files the vector components are in the directions of the axes in the Global system. Thus for example the heave motion is the motion vertical to the sea surface and not the motion along the z-axis in the User system. Hence for example the vertical bending moment is not strictly speaking comparable between two different runs if the trim angle is not the same. This should also be kept in mind when comparing with results from a FEM analysis where the results will refer to the ship fixed User system.

3.4.3 Frequency domain output

The module *Wasim_Fourier* transforms data from time domain to frequency domain. A large number of time domain computations can be included in the analysis to produce transfer functions. In addition to the listing file, which contains some statistical data for the time series, the following files are written:

- fourierG1.SIF

  This file can be read into Postresp for display of results and statistical processing in frequency domain. Transfer functions for the following quantities are included (if time domain results are available):

  — rigid body motion at the motion reference point

  — total force and moment on vessel (appears as EXFORC in Postresp) (moments with respect to motion reference point)

— sectional loads

— selected pressures

— relative motion (appears as ELEV in Postresp)

Transfer functions for all wave periods and headings and all vessel speeds are included in the file. For the statistical computations to work the transfer functions must be given for the same frequency set for all wave headings. This is secured by interpolating all data onto a fixed set of wave frequencies equally spaced between the smallest and largest frequency in the dataset. If the user has given identical frequencies for all headings these frequencies can optionally be used as they are (cf. Chapter 3.3).

Note: The vector results on this file has components along the axes of the Global system.

- loadG1.SIF
  This file is input to the load transfer module *Wasim_Stru*. The file contains the pressures on all panels and the rigid body motion for all load cases. Each frequency/heading pair is a load case. For the load transfer to work properly, the wave frequencies must be identical for all headings. This can be obtained by interpolating data onto a uniformly distributed dataset. It is, however, recommended to use an input dataset in the time domain analysis with the same frequencies for all headings as the interpolation is known to potentially introduce load imbalance.

Note: The data on this file are to be carried over to a FEM model. However, the displacement and acceleration data still refer to the motion reference point, and the vector components are in the directions of the axes in the Global system.

- fourier.fmout
  This file contains tabulated transfer functions for the rigid body motion and/or the force and moment. The results are always presented for the original frequencies used in the time domain analysis.

- fourier.ldout
  This file contains tabulated transfer functions for the sectional loads at the original frequencies used in the time domain analysis.

- fourier.relout
  This file contains tabulated transfer functions for the relative motion at the original frequencies (i.e. the frequencies used in the time domain analysis).

- fourier.prout
  This file contains tabulated transfer functions for the selected pressures at the original frequencies used in the time domain analysis.

3.4.4 Load transfer output

The module *Wasim_Stru* performs the load transfer from the panel model used in Wasim to the FEM model used in SESAM:Sestra. The module produces the following output files (in addition to a listing file):

- <prefix>SN.FEM
  N is the top level superelement. The file is used as input to SESAM:Sestra. It is only written in the case of frequency domain load transfer. This file is necessary if a fatigue analysis by SESAM:Stofat is to be carried out after running SESAM:Sestra. The prefix will be the same as the prefix on the corresponding T-files. This prefix is the path to the files so the output files will also end up in this directory.

- <prefix>L*.FEM
  These files are output from *Wasim_Stru* and contain the loads on the FEM model to be used as input to SESAM:Sestra. The prefix will be the same as the prefix on the corresponding T-files.

3.5 Program Requirements

3.5.1 Execution Time

All the modules in the system, except for the preprocessor require significant execution time, but *Wasim_Solve* is by far the most demanding. The computational time required depends heavily on the number of panels, number of time steps, number of wave directions etc. The CPU-effort for a complete linear analysis will typically be in the range 10-100 hours on a medium/low-end PC (2Ghz Pentium IV).

For an individual time series analysis the ratio of CPU-time over real time is (on the same PC) in most cases in the range 3-20.

3.5.2 Memory

All modules are working in-core for maximum efficiency. This puts requirements on the computers to be used. As for the CPU-time *Wasim_Solve* is normally the most demanding. However, for load transfer *Wasim_Fourier* may also require a lot of memory. It is not recommended to run Wasim on computers with less than 512MB memory. For complex problems (e.g. multihull vessel, very high speed vessels) at least 1 GB is needed.

3.6 Program Limitations

File names

Blanks are not allowed in file names including paths references as a part of the name. File names including path cannot exceed 200 characters, the primary input file name cannot have more than 80 characters including the path.

The paths and file names will be automatically created by the Wasim Manager. Relative paths are used so the path to the workspace directory may contain blanks.

Rudders

A maximum of 6 rudders can be defined.

Model limitations

The geometry input file cannot have more than 200 stations and each station cannot have more than 200 points.

Only one vessel can be included in each workspace for the *Wasim Manager*

Load transfer

Static loads cannot be transferred together with frequency domain loads. Static loads may be transferred by running a calm sea run and performing a snapshot load transfer at a point in time when steady state has been obtained.

Other limitations

The total number of sections (i.e. the sum of sections normal to x-axis, y-axis and z-axis) must not exceed 100.

Maximum number of wave components: 1000.

Maximum number of points for computing relative motion: 40.

Maximum number of selected pressures: 100.

Maximum number of point masses: 140000.